

# Modern Engineering

January 12 | Day 5

## Prudential Financial



# Whilst we wait....

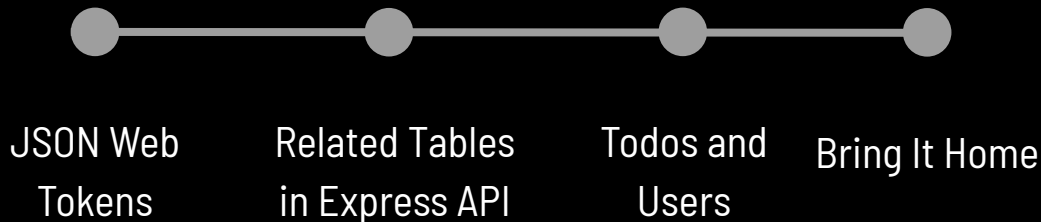
**In chat write:**

**A book/movie/TV series that will blow our mind.**

**Let's give everyone  
a min or two to join**



# LESSON ROADMAP



# MEF MODULE 1 DAY 5: JWTs and Authenticating APIs

Schedule	
9:00–9:15 am	Welcome and Warm-Up
9:15–10:00 am	<b>JSON Web Tokens</b>
10:00–12:30 am	<b>Authentication with JWTs Walkthrough</b>
12:30–1:30 pm	Lunch
1:30–2:30 pm	<b>Related Models: Users and To Dos</b>
2:30 - 4:50 pm	<b>Express API Review and Working Session</b>
4:50–5:00 pm	Bring It Home



# LEARNING OBJECTIVES

1

Identify the use case for JSON Web Tokens when securing a microservices API

2

Send requests and parse responses encrypted with JSON Web Tokens

3

Understand how JWTs enable authentication and authorization in an Express API

4

Protect routes in an Express API based on tokens



Modern Engineering



# Security with JSON Web Tokens



# Recap: Authentication and Authorization

## Authentication

You are who you say you are.

Including a header in HTTP requests.

**VS.**

## Authorization

You are permitted to perform this operation.

Including role fields in the user database and "user" fields to track owners of data.



## Guided Walk-Through: Auth with JWT



Let's hop over to GitHub to dig deeper into a popular, lightweight approach to authentication/authorization - JSON Web Tokens.

<https://bit.ly/45B1N3t>





Modern Engineering



# Express JWT Authentication



# Lunch Break





Let's add a Users table to our To-Do App and associate users with given to-do items.

[We'll follow the steps outlined in this git repository.](#)



**Solo Exercise:**

# Adding Users to Your App

90 minutes



Let's practice this relationship pattern by adding a similar users table to your own Express API.

As a bonus, you could beef up the authentication and add a login route that returns a JWT on successful matching of a username and password. This JWT would be included in the user's future HTTP requests as a header, and your express app would decode this token for any routes requiring an authenticated user.





To wrap up this section of the course, let's use the remaining time to review what we've done so far, provide answers to any questions you have, and demonstrate any bonus authentication features we have time for!

We won't be requiring authentication in future applications during this course, but it's a great bonus feature to consider once you've reached minimum requirements on future projects!



## Solo Exercise: Daily Exit Ticket

3 minutes



- Please take a moment to give us your feedback after today's course!
- Use the QR code or follow the link: <https://bit.ly/pruMEFc2>



**Scan Me!**